COMP 2401 Review – Fall 2013

- Introduction
  - Basic programming concepts
  - Computer organization
    - Hardware, OS, applications
    - Virtual memory (code segment, data segment (heap), function call stack)

- Data representation
  - Primitive data types
    - Bit models (two's complement, magnitude-only, fixed point, floating point, ASCII)
    - Bitwise operations (left/right shift, and, or, not, xor)
    - Non-decimal bases
  - Compound data types
    - Arrays, strings
    - Structures (pointers to structures, passing by value/reference)
  - Pointers
    - Operators: dereferencing (*), address-of (&)
    - Pass-by-value and pass-by-reference
    - Declare pointers, using, initializing
    - Arrow operator vs. dot operator
    - Double pointers:
      - Pass a pointer by reference
      - Array of pointers
    - Pointer arithmetic
      - Used for arrays
    - Command line arguments

- Memory management
  - Stack and heap
    - Areas of virtual memory
    - Stack keeps track of function calls and returns
    - Heap stores dynamically allocated variables
  - Dynamic memory allocation
    - Sometimes we don't know at compile time how much memory we'll need
    - malloc, calloc, free, realloc
  - Linked lists
    - pointer practice
    - advanced linked lists: separate the nodes (list-specific info) from the data (application-specific info)
    - add first element, add to front, add to back, add to middle, removing element, searching a list, traversing a list, freeing (nodes, sometimes data)

- Program structure
  - building: compiling, linking, Makefiles, assembly code
  - design procedural programs: make modular reusable functions
  -

    - code organization
        - separating header and source files
        - scope (block, file)
        - storage class (static, automatic, extern)

- Libraries
    - how to make a library
        - naming conventions
    - how to use a library

- I/O
    - streams, data sources and sinks
    - files, devices
    - buffering (line buffering, block buffering, unbuffered)
    - pipes, redirection

- Concurrent computing
    - multi-threaded, multi-process, distributed
    - processes
        - process management (put process in background, suspended a process, put it in foreground, fork, exec, shell commands)
        - signals (send a signal, install signal handler)
        - sockets (TCP, UDP, client-server architecture)
    - threads
        - flow of control in a process
        - share data segment (heap), code segment, virtual memory
        - concurrent data modification, race conditions
        - semaphores (mutex)

Final exam !!! Dec. 16

- 3 hours
- Concepts and exercises: 50 marks (8 questions)
- Programming: 50 marks (3 questions)
- 1 bonus question: 3 marks

Study tips:

- concepts: buddy-study (make up questions)
- programming: program! redo A3 and A4, and Test #1 and #2